Raymond Lin
304937942

CS M146 - Problem Set 2

1. a) OR inputs $x \in \mathbb{R}^2$, outputs $y \in \{-1, +1\}$

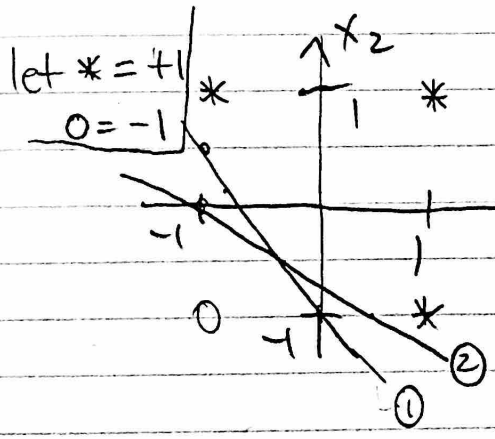
	x_1	x_2	y	
①	-1	-1	-1	$w_1 x_1 + w_2 x_2 + b = 0$
②	-1	+1	+1	$w_1 x_1 + w_2 x_2 = -b$
③	+1	-1	+1	
④	+1	+1	+1	

① $w_1(-1) + w_2(-1) < b \Rightarrow -1$

② $w_1(-1) + w_2(+1) > b \Rightarrow +1$

③ $w_1(+1) + w_2(-1) > b \Rightarrow +1$

④ $w_1(+1) + w_2(+1) > b \Rightarrow +1$



y-intercept form:

$$x_2 = -\left(\frac{w_1}{w_2}\right) x_1 - \frac{b}{w_2}$$

let $w_1 = 1.5$, $w_2 = 1$, $b = 1$

let $w_1 = 2$, $w_2 = 3$, $b = 2$

Two possible solutions

① $1.5 x_1 + x_2 + 1 = 0$

② $2x_1 + 3x_2 + 2 = 0$

b) XOR inputs $x \in R^2$, outputs $y \in \{-1, +1\}$

	x_1	x_2	y	
①	-1	-1	-1	$\rightarrow w_1(-1) + w_2(-1) < b$
②	-1	+1	+1	$\rightarrow w_1(-1) + w_2(+1) > b$
③	+1	-1	+1	$\rightarrow w_1(+1) + w_2(-1) > b$
④	+1	+1	-1	$\rightarrow w_1(+1) + w_2(+1) < b$

① + ④ $-w_1 + w_1 - w_2 + w_2 < b + b$, $0 < 2b$

② + ③ $-w_1 + w_1 - w_2 + w_2 > b + b$, $0 > 2b$

} contradiction.

2. $$J(\theta) = -\sum_{n=1}^{N} \left[y_n \log(h_\theta(x_n)) + (1-y_n) \log(1-h_\theta(x_n)) \right]$$

a) $$\frac{\partial}{\partial \theta_j} J(\theta) = -\sum_n \underbrace{\frac{\partial}{\partial \theta_j} (y_n \log(h_\theta(x_n)))}_{A} + \underbrace{\frac{\partial}{\partial \theta_j} ((1-y_n) \log(1-h_\theta(x_n)))}_{B}$$

- we know:
$$h_\theta(x_n) = \sigma(\theta^T x_n)$$

$$A = \frac{\partial}{\partial \theta_j} (y_n \log(\sigma(\theta^T x_n)))$$
$$= y_n \cdot \frac{1}{\sigma(\theta^T x_n)} \cdot \frac{\partial}{\partial \theta_j} \sigma(\theta^T x_n)$$

$$B = \frac{\partial}{\partial \theta_j} ((1-y_n) \log(1-\sigma(\theta^T x_n)))$$
$$= (1-y_n) \cdot \frac{1}{1-\sigma(\theta^T x_n)} \cdot \frac{\partial}{\partial \theta_j} (1-\sigma(\theta^T x_n))$$
$$= -\frac{1-y_n}{1-\sigma(\theta^T x_n)} \cdot \frac{\partial}{\partial \theta_j} \sigma(\theta^T x_n)$$

- x_n is a constant
- let $a = \theta^T x_n$
- then $\frac{\partial}{\partial \theta_j} \sigma(\theta^T x_n) = \frac{d}{da} \sigma(a)$

$$\sigma(a) = \frac{1}{1+e^{-a}}$$

$$\frac{d}{da} \sigma(a) = -\frac{1}{(1+e^{-a})^2} \cdot e^{-a} \cdot -1 = \frac{e^{-a}}{(1+e^{-a})^2}$$
$$= \frac{1}{1+e^{-a}} \cdot \frac{e^{-a}}{1+e^{-a}}$$
$$= \frac{1}{1+e^{-a}} \cdot \left(\frac{1+e^{-a}}{1+e^{-a}} - \frac{1}{1+e^{-a}} \right) = \sigma(a)(1-\sigma(a))$$

$x_n = \begin{bmatrix} x_{n1} \\ x_{n2} \\ \vdots \\ x_{nd} \end{bmatrix}$ ← features values of x_n

data pt.

Thus $\frac{\partial}{\partial \theta_j} \sigma(\theta^T x_n) = \sigma(a)(1-\sigma(a))$

$= \sigma(\theta^T x_n)(1-\sigma(\theta^T x_n)) x_n$

- then

$A = \frac{y_n}{\sigma(\theta^T x_n)} \cdot \sigma(\theta^T x_n)(1-\sigma(\theta^T x_n)) x_n$

$= y_n (1-\sigma(\theta^T x_n)) x_n$

$B = -\frac{(1-y_n)}{(1-\sigma(\theta^T x_n))} \cdot \sigma(\theta^T x_n)(1-\sigma(\theta^T x_n)) x_n$

$= -(1-y_n)(\sigma(\theta^T x_n)) x_n$

$= -\sigma(\theta^T x_n) x_n + y_n \sigma(\theta^T x_n) x_n$

- then

$\frac{\partial}{\partial \theta_j} J(\theta) = -\sum_n^N \left(y_n x_n - y_n \sigma(\theta^T x_n) x_n + \sigma(\theta^T x_n) x_n - y_n \sigma(\theta^T x_n) x_n \right)$

$= -\sum_n^N (y_n - \sigma(\theta^T x_n)) x_n$

$$\boxed{\frac{\partial}{\partial \theta_j} J(\theta) = -\sum_n^N (y_n - h_\theta(x_n)) x_{nj}}$$

b) $\frac{\partial^2}{\partial \theta_j \partial \theta_k} J = \frac{\partial}{\partial \theta_k} \left(\frac{\partial}{\partial \theta_j} J(\theta) \right)$

$= \frac{\partial}{\partial \theta_k} \left(-\sum_n^N (y_n - h_\theta(x_n)) x_{nj} \right)$

$= -\sum_n^N x_{nj} \left(\frac{\partial}{\partial \theta_k} (y_n - h_\theta(x_n)) \right)$

$= -\sum_n^N x_{nj} \left(\frac{\partial}{\partial \theta_k} (y_n - \sigma(\theta^T x_n)) \right) = -\sum_n^N x_{nj} \left(-\frac{\partial}{\partial \theta_k} \sigma(\theta^T x_n) \right)$

Note: vectors are usually column vectors

- we know

$$\frac{\partial}{\partial \theta_k} \sigma(\theta^T x_n) = \sigma(\theta^T x_n)\left(1 - \sigma(\theta^T x_n)\right) \cdot x_{nk}$$

- we have

$$-\sum_{n}^{N} y_{nj}\left(-\sigma(\theta^T x_n)\left(1 - \sigma(\theta^T x_n)\right) \cdot x_{nk}\right.$$

$$= \sum_{n}^{N} \sigma(\theta^T x_n)\left(1 - \sigma(\theta^T x_n)\right) x_{nj} x_{nk}$$

$$\frac{\partial^2 J(\theta)}{\partial \theta_k \partial \theta_j} = \sum_{n=1}^{N} \sigma(\theta^T x_n)\left(1 - \sigma(\theta^T x_n)\right) x_{nj} x_{nk}$$

$$= \sum_{n=1}^{N} h_\theta(x_n)\left(1 - h_\theta(x_n)\right) x_{nj} x_{nk}$$

- then

$$H = \sum_{n=1}^{N} h_\theta(x_n)\left(1 - h_\theta(x_n)\right) x_n x_n^T$$

□

c) - We want to show that J is a convex function,
- We can show that J's Hessian is positive semi definite
- We know

$$H = \sum_n^N h_\theta(x_n)\,(1-h_\theta(x_n))\,x_n x_n^T$$

- We know that H is positive semi-definite iff

$$z^T H z = \sum_{j,k} z_j z_k H_{jk} \geq 0$$

for all z real vectors

- We know that $h_\theta(x_n) = \sigma(\theta^T x_n)$

and

$$0 \leq \sigma(\theta^T x_n) \leq 1$$

so $$0 \leq h_\theta(x_n) \leq 1$$

- Now

$$H = \sum_n^N h_\theta(x_n)\,(1-h_\theta(x_n))\,x_n x_n^T$$

$0 \leq h_\theta(x_n) \leq 1$ (positive) $\quad 0 \leq 1 - h_\theta(x_n) \leq 1$ (positive)

- let $h_\theta(x_n)(1-h_\theta(x_n)) = C$ ← some positive constant

$$H = C \sum_n^N x_n x_n^T$$

$$H = C\,XX^T$$

$$z^T H z = C(z^T X X^T z)$$

$$= C\left((X^T z)^T (X^T z)\right)$$

(positive) (positive)

← any matrix transpose multiplied with itself is basically squaring all its values

so $z^T H z \geq 0$

∴ H is positive semi-definite
∴ J(θ) is a convex function □

3. $\hat{\theta}_{MLE} = \text{argmax}_\theta (L(\theta))$

a) $L(\theta) = P(X_1, \ldots, X_n; \theta)$
$= P(X_1, \theta) \cdot P(X_2, \theta) \cdot \ldots \cdot P(X_n, \theta)$
$= \theta \cdot (1-\theta) \cdot \ldots$

$L(\theta) = \theta^{x_i} \cdot (1-\theta)^{(1-x_i)}$
$= \prod_n^N P(y_n | x_n; \theta)$
- does not depend on order, because variables are independent of each other

b) $\ell(\theta) = \log(L(\theta))$
$= \log\left(\prod^N P(y_n | x_n; \theta)\right)$
$= \log\left(\prod^N \theta^{x_i} \cdot (1-\theta)^{(1-x_i)}\right)$
$= \sum^N \left(\log(\theta^{x_i} \cdot (1-\theta)^{(1-x_i)}\right)$
$= \sum^N \left(\log(\theta^{x_i}) + \log\left((1-\theta)^{(1-x_i)}\right)\right)$
$= \sum^N \left(x_i \log(\theta) + (1-x_i)\log(1-\theta)\right)$

$\frac{d}{d\theta}\ell(\theta) = \frac{d}{d\theta} \sum_i^N \left(x_i \log(\theta) + (1-x_i)\log(1-\theta)\right)$
$= \sum_i^N \frac{d}{d\theta}(x_i \log(\theta)) + \frac{d}{d\theta}(1-x_i)\log(1-\theta)$
$= \sum_i^N \frac{x_i}{\theta} - \frac{1-x_i}{1-\theta}$

$\frac{d^2}{d\theta^2}\ell(\theta) = \frac{d}{d\theta}\left(\sum_i^N \frac{x_i}{\theta} - \frac{1-x_i}{1-\theta}\right)$
$= \sum_i^N \left(\frac{d}{d\theta}\frac{x_i}{\theta} - \frac{d}{d\theta}\frac{1-x_i}{1-\theta}\right)$
$= \sum_i^N \left(-\frac{x_i}{\theta^2} - \frac{1-x_i}{(1-\theta)^2}\right)$

- MLE: Set first derivative equal to 0

$$\frac{d}{d\theta}\ell(\theta) = \sum_{i}^{N}\left(\frac{x_i}{\theta} - \frac{1-x_i}{1-\theta}\right) = 0$$

$$= \sum_{i}^{N}\left(\frac{x_i(1-\theta) - \theta(1-x_i)}{\theta(1-\theta)}\right)$$

$$= \sum_{i}^{N}\left(\frac{x_i - \theta x_i - \theta + \theta x_i}{\theta(1-\theta)}\right)$$

$$= \sum_{i}^{N}\left(\frac{x_i - \theta}{\theta(1-\theta)}\right) = 0$$

$$\sum_{i}^{N} x_i - N\theta = 0$$

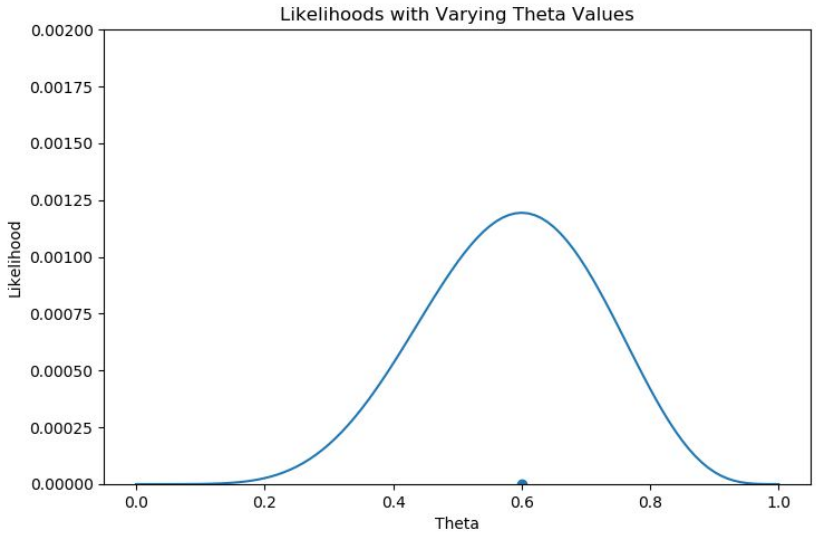
$$N\theta = \sum_{i}^{N} x_i$$

$$\boxed{\theta = \frac{\sum_{i}^{N} x_i}{N}}$$

3.

c)

Graph:



Closed Form Answer:

$$L(\theta) = \theta^6(1-\theta)^4$$

$$\frac{dL}{d\theta} = 6\theta^5(1-\theta)^4 + 4\theta^6(1-\theta)^3(-1)$$

$$= 6\theta^5(1-\theta)^4 - 4\theta^6(1-\theta)^3$$

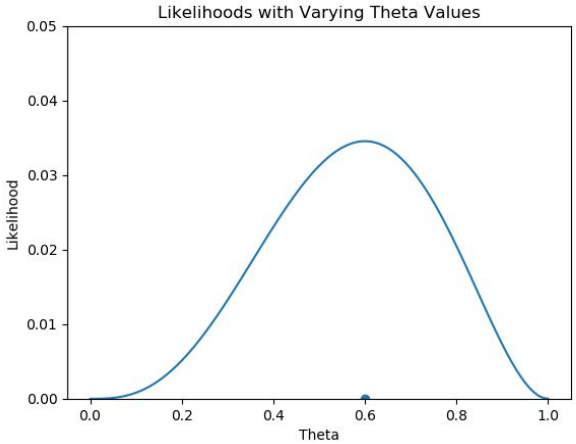
$$= (2\theta^5(1-\theta)^3)(3(1-\theta) - 2\theta)$$

$$(2\theta^5)(1-\theta)^3(3-5\theta) = 0$$

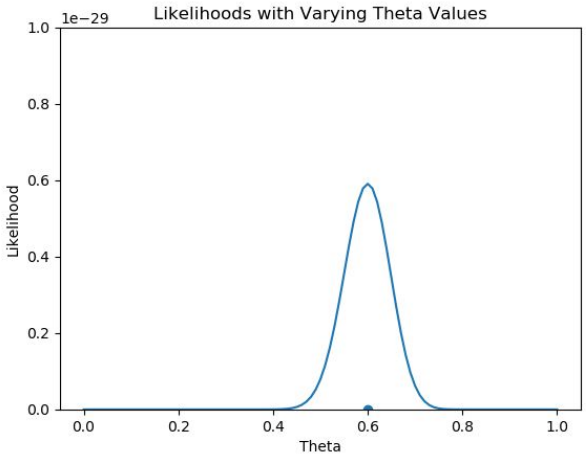
$$\theta = \frac{3}{5}$$

Yes, the graph agrees with the closed form answer.

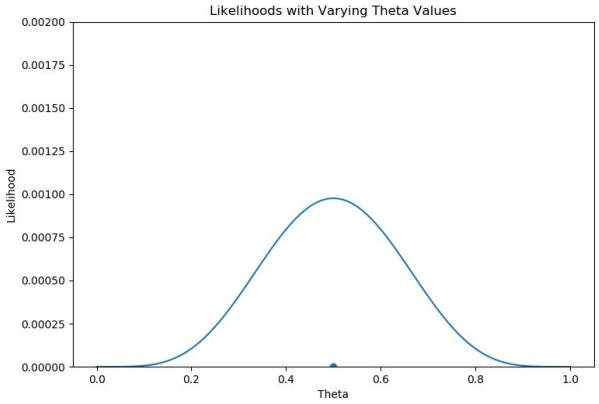
d) $n = 5$, *successes* $= 3$



$n = 100,\ successes\ =\ 60$



$n = 10,\ successes\ =\ 5$



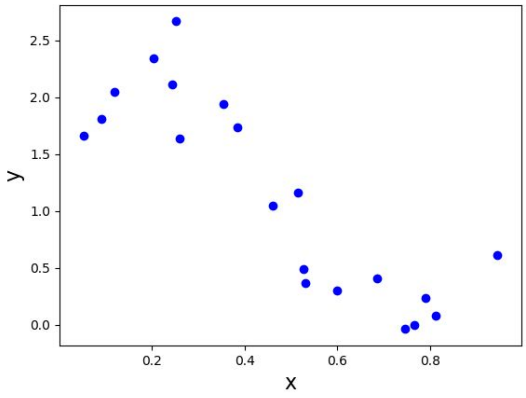
The maximum likelihood estimates are approximately equal to the mean of the dataset. For example, when there are 5 trials and 3 successes, the maximum likelihood estimate and mean are both equal to ⅗ or 0.6. We also notice that the amplitude of the likelihood function increases as n increases. This is due to the law of large numbers, which states that as the number of samples increases, the average of the trials is more likely to be close to the mean or expected value. Variance also decreases with more trials. Thus, the experiments with higher n have a taller and skinnier graph, while the experiments with lower n have a shorter and wider graph.

4.

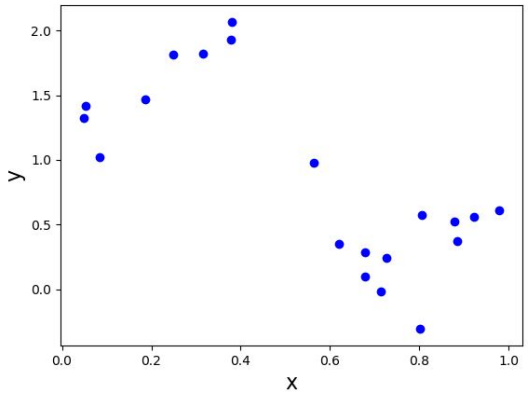
Linear Regression

a)

Train



Test



The data looks as though it can follow a general linear trend. It follows an inverse relationship. As the x-values increase, the y-values decrease. Thus, linear regression will be effective on this dataset.

b)

done

c)

done

d)

Gradient Descent

Step Size (η)	**Coefficient 1**	**Coefficient 2**	**Iterations**	**Cost (J(θ))**	**Time (s)**
0.01	2.44640703	-2.81635346	764	3.91257641	0.021
0.001	2.4464068	-2.816353	7020	3.91257641	0.155
0.0001	2.27055798	-2.46064834	10000	4.08639704	0.214
0.0407	-9.40471e+18	-4.65229e+18	10000	2.71092e+39	0.262

It appears that the most accurate coefficients are obtained by using a step size of 0.01 and 0.001. Using these step sizes, the algorithm actually converges, as the number of iterations is less than the maximum allowed iterations. For a step size of 0.0001, it appears that the algorithm was on

its way to converging, until the maximum iterations stopped the algorithm. As for cost and time, as the step size decreased, the cost and elapsed time increased. It may seem that a larger step size is better, but this is not the case. For a step size of 0.0407, the coefficients are completely off, and the cost and elapsed time are the highest, even though it reached the maximum number of iterations. This is because the step size was too big, so the algorithm just went back and forth endlessly.

e)

Closed Form Solution

Coefficient 1	**Coefficient 2**	**Cost (J(θ))**	**Time (s)**
2.44640709	-2.81635359	3.91257640579	0.003

We achieve similar coefficients as gradient descent, with only a small margin of error. Cost is also the same. In contrast, the closed-form solution runs faster than that of gradient descent. This is probably because we have a small dataset. If the dataset were much larger, this method would probably be slower.

f)

Gradient Descent with Step Size as a Function of Current Iteration

Coefficient 1	**Coefficient 2**	**Cost (J(θ))**	**Time (s)**
2.44640678	-2.816353596	3.91257640579	0.043

Using a step size that is a function of the current iteration, the algorithm runs in about 0.043 seconds. This is slightly slower than the best parameter for gradient descent. Cost remains the same.

Polynomial Regression

g)

done

h)

Root mean square error is more desirable than regular mean square error, because it scales the error back down to the size of the error, instead of the square of the size. If we used regular mean square error, outliers would affect the final error by a factor of the square of its value. In other words, outliers would have a much larger impact on the final error value. In contrast, outliers in root mean square error would just affect the final error by a factor of its original value.

i)

It appears that a polynomial of degree 5 fits the data best. At degree 5, both the train and test set errors are the lowest. If we increase the degree to 6, we can see that the test set error increases. If we decrease the degree to 4, we can see that both the train and test set errors increase very slightly. There is evidence of both underfitting and overfitting. For polynomials of degree 0, 1 and 2, the data is being underfitted, because the error rates are high, and the degree value of the polynomial is not capturing the trend of the data very well. For polynomials of degree 6 and above, we can see overfitting, because the test set errors increase, while the train set errors remain constant. This becomes obvious at degree 9 and above.